

# CHAPTER 2: MODELING DATA IN THE

**Modern Database Management**  
**11<sup>th</sup> Edition**

**Jeffrey A. Hoffer, V. Ramesh,  
Heikki Topi**

---

# OBJECTIVES

---

- ✖ Define terms
- ✖ Understand importance of data modeling
- ✖ Write good names and definitions for entities, relationships, and attributes
- ✖ Distinguish unary, binary, and ternary relationships
- ✖ Model different types of attributes, entities, relationships, and cardinalities
- ✖ Draw E-R diagrams for common business situations
- ✖ Convert many-to-many relationships to associative entities
- ✖ Model time-dependent data using time stamps

# A GOOD DATA NAME IS:

---

- ✖ Related to business, not technical, characteristics
- ✖ Meaningful and self-documenting
- ✖ Unique
- ✖ Readable
- ✖ Composed of words from an approved list
- ✖ Repeatable
- ✖ Written in standard syntax



# DATA DEFINITIONS

---

- ✗ Explanation of a term or fact
  - + Term—word or phrase with specific meaning
  - + Fact—association between two or more terms
- ✗ Guidelines for good data definition
  - + A concise description of essential data meaning
  - + Gathered in conjunction with systems requirements

Chapter 2 Data Definitions  
Copyright © 2010 Pearson Education, Inc. Publishing as Prentice Hall



# E-R MODEL CONSTRUCTS

---

## ✖ Entities:

- + Entity instance—person, place, object, event, concept (often corresponds to a row in a table)
- + Entity Type—collection of entities (often corresponds to a table)

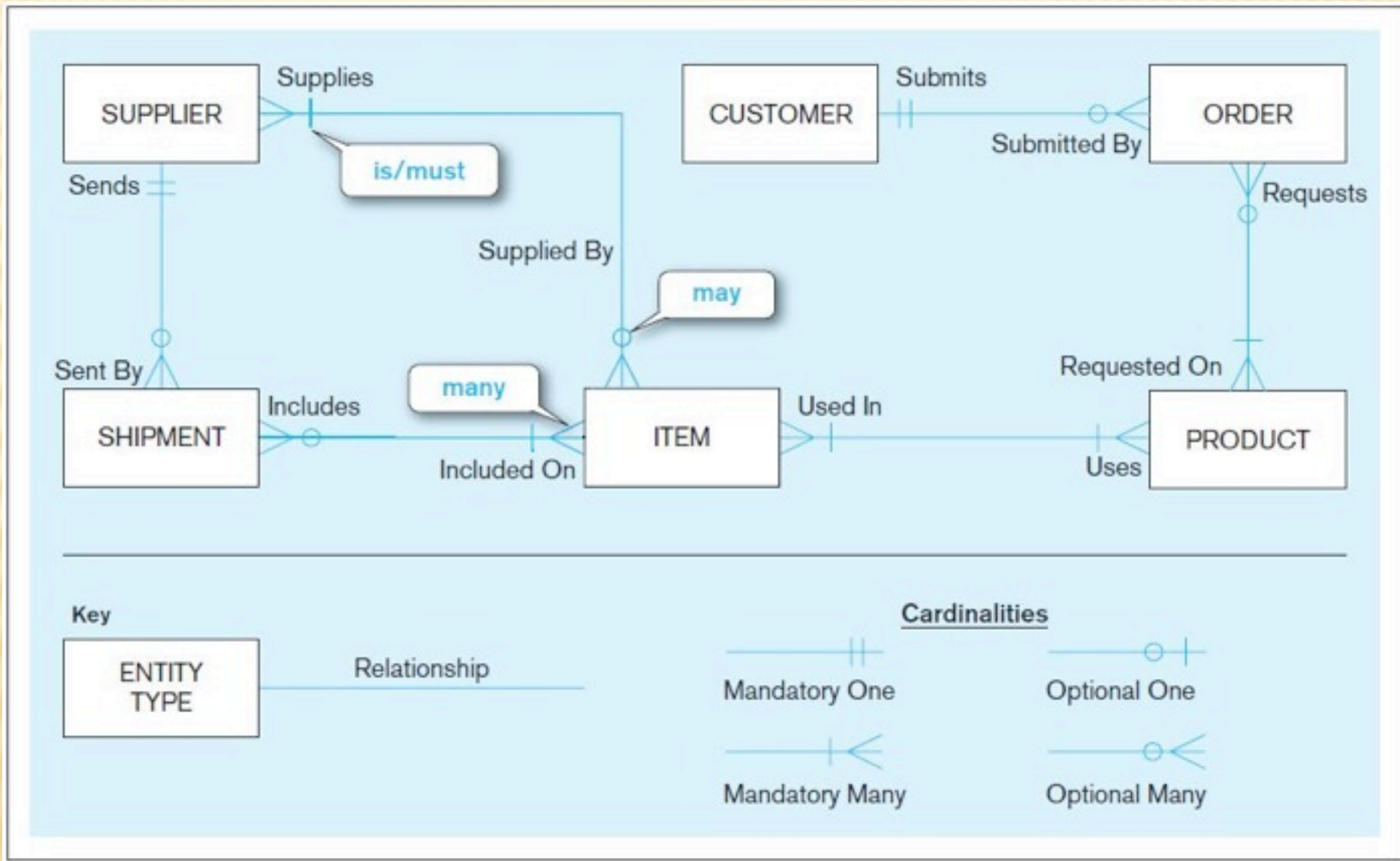
## ✖ Relationships:

- + Relationship instance—link between entities (corresponds to primary key–foreign key equivalencies in related tables)
- + Relationship type—category of relationship...link between entity types

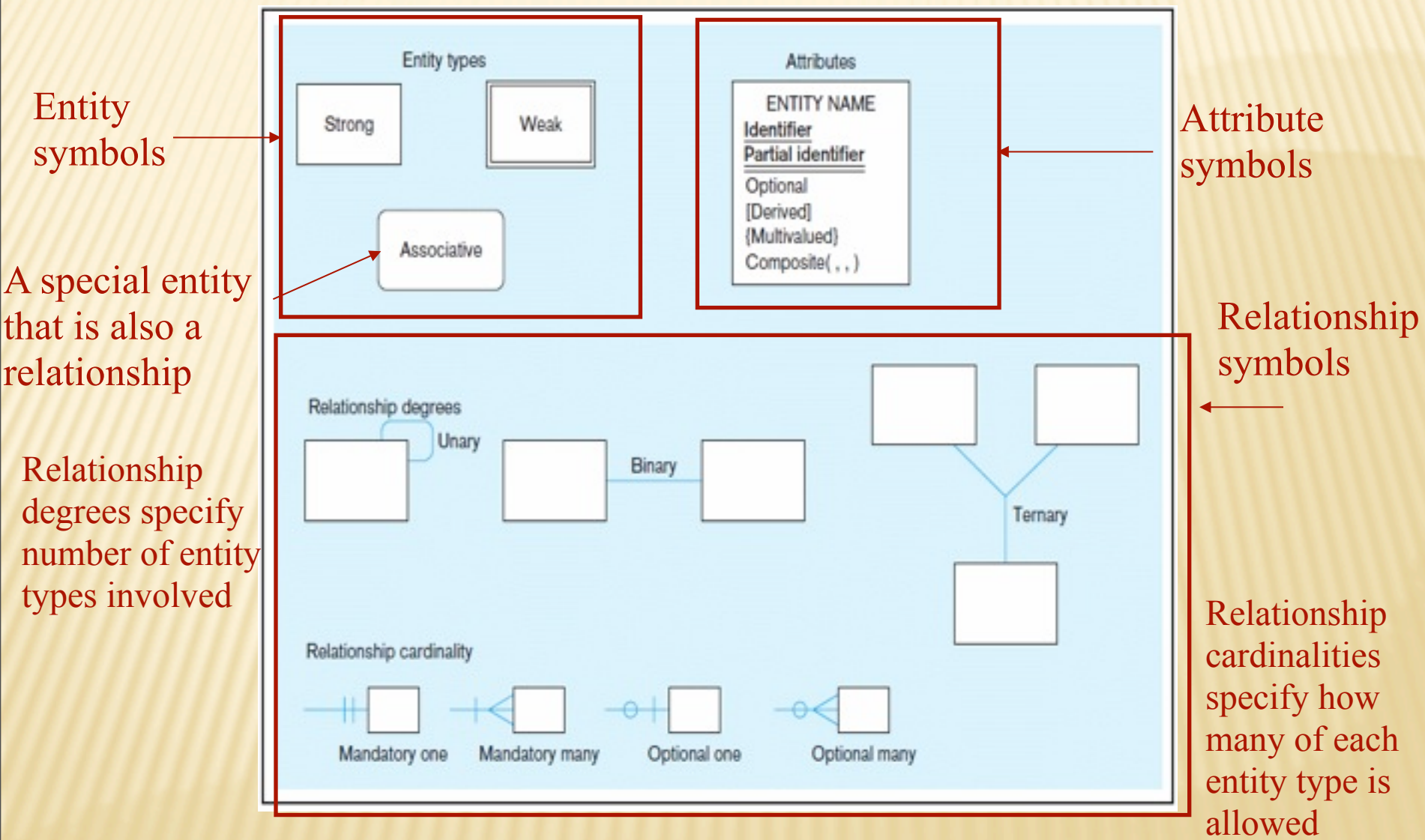
## ✖ Attributes:

- + Properties or characteristics of an entity or relationship type (often corresponds to a field in a table)

# Sample E-R Diagram (Figure 2-1)



# Basic E-R notation (Figure 2-2)





# BUSINESS RULES

---

- ✖ Are statements that define or constrain some aspect of the business
- ✖ Are derived from policies, procedures, events, functions
- ✖ Assert business structure
- ✖ Control/influence business behavior
- ✖ Are expressed in terms familiar to end users
- ✖ Are automated through DBMS software

# A GOOD BUSINESS RULE IS:

---

- ✗ Declarative–what, not how
- ✗ Precise–clear, agreed–upon meaning
- ✗ Atomic–one statement
- ✗ Consistent–internally and externally
- ✗ Expressible–structured, natural language
- ✗ Distinct–non–redundant
- ✗ Business–oriented–understood by

# ENTITIES

---

- ✗ **Entity** – a person, a place, an object, an event, or a concept in the user environment about which the organization wishes to maintain data
- ✗ **Entity type** – a collection of entities that share common properties or characteristics
- ✗ **Entity instance** – A single occurrence of an entity type



# ENTITY TYPE AND ENTITY

Entity type: EMPLOYEE			
Attributes	Attribute Data Type	Example Instance	Example Instance
Employee Number	CHAR (10)	642-17-8360	534-10-1971
Name	CHAR (25)	Michelle Brady	David Johnson
Address	CHAR (30)	100 Pacific Avenue	450 Redwood Drive
City	CHAR (20)	San Francisco	Redwood City
State	CHAR (2)	CA	CA
Zip Code	CHAR (9)	98173	97142
Date Hired	DATE	03-21-1992	08-16-1994
Birth Date	DATE	06-19-1968	09-04-1975

**FIGURE 2-3** Entity type EMPLOYEE with two instances

# AN ENTITY...

---

## ✗ SHOULD BE:

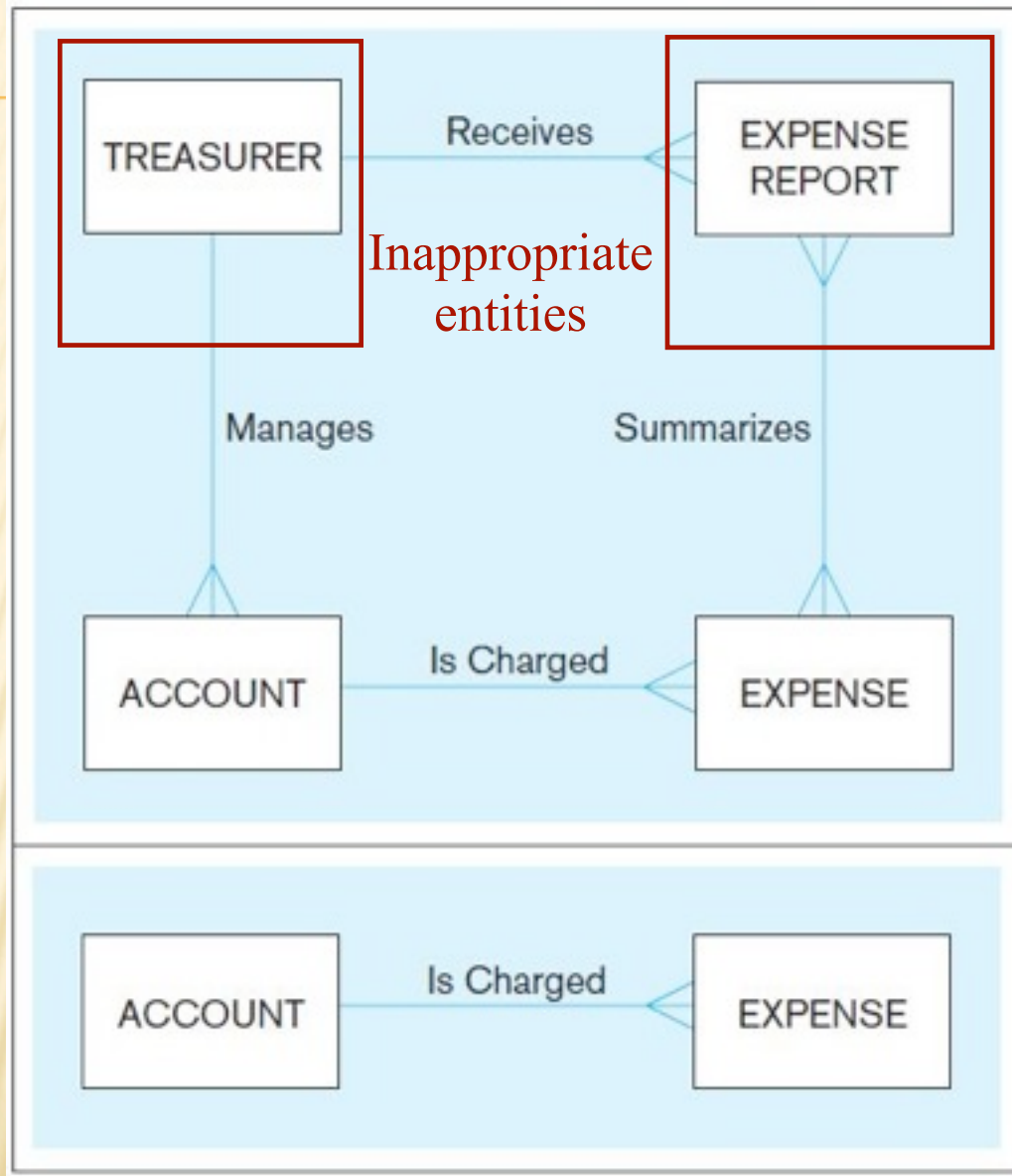
- + An object that will have many instances in the database
- + An object that will be composed of multiple attributes
- + An object that we are trying to model

## ✗ SHOULD NOT BE:

- + A user of the database system
- + An output of the database system (e.g., a report)

Figure 2-4 Example of inappropriate entities

**System  
user**



**System output**

**Appropriate  
entities**



# STRONG VS. WEAK ENTITIES, AND IDENTIFYING RELATIONSHIPS

## ✗ Strong entity

- + exists independently of other types of entities
- + has its own unique identifier
  - ✗ identifier underlined with single line

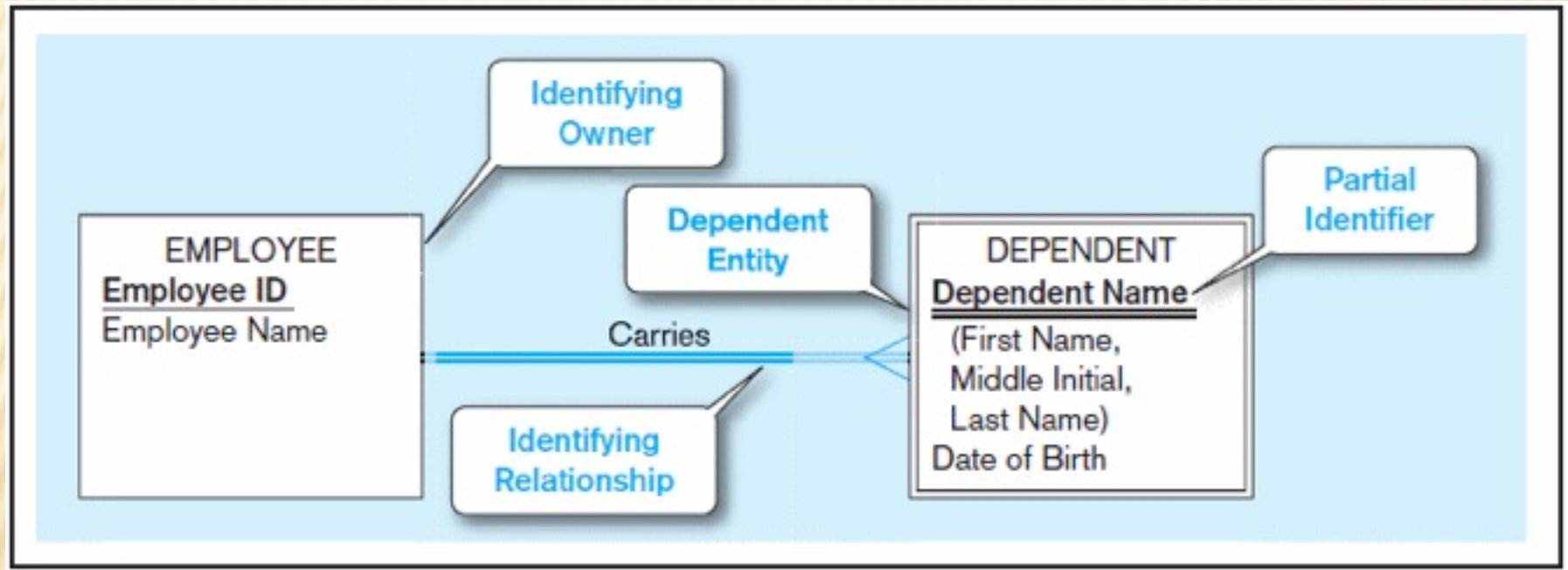
## ✗ Weak entity

- + dependent on a strong entity (identifying owner)... cannot exist on its own
- + does not have a unique identifier (only a partial identifier)
- + entity box and partial identifier have double lines

## ✗ Identifying relationship

- + links strong entities to weak entities

Figure 2-5 Example of a weak identity and its identifying relationship



Strong entity

Weak entity

# ATTRIBUTES

---

- ✗ Attribute–property or characteristic of an entity or relationship type
- ✗ Classifications of attributes:
  - + Required versus Optional Attributes
  - + Simple versus Composite Attribute
  - + Single–Valued versus Multivalued Attribute
  - + Stored versus Derived Attributes

Chapter 10 © 2011 Pearson Education, Inc. Publishing as Prentice Hall



# REQUIRED VS. OPTIONAL ATTRIBUTES

Entity type: STUDENT				
Attributes	Attribute Data Type	Required or Optional	Example Instance	Example Instance
Student ID	CHAR (10)	Required	876-24-8217	822-24-4456
Student Name	CHAR (40)	Required	Michael Grant	Melissa Kraft
Home Address	CHAR (30)	Required	314 Baker St.	1422 Heft Ave
Home City	CHAR (20)	Required	Centerville	Miami
Home State	CHAR (2)	Required	OH	FL
Home Zip Code	CHAR (9)	Required	45459	33321
Major	CHAR (3)	Optional	MIS	

**Required** – must have a value for every entity (or relationship) instance with which it is associated

**Optional** – may not have a value for every entity (or relationship) instance with which it is associated

# SIMPLE VS. COMPOSITE ATTRIBUTES

- ✦ **Composite attribute** – An attribute that has meaningful component parts

The address is broken into component parts

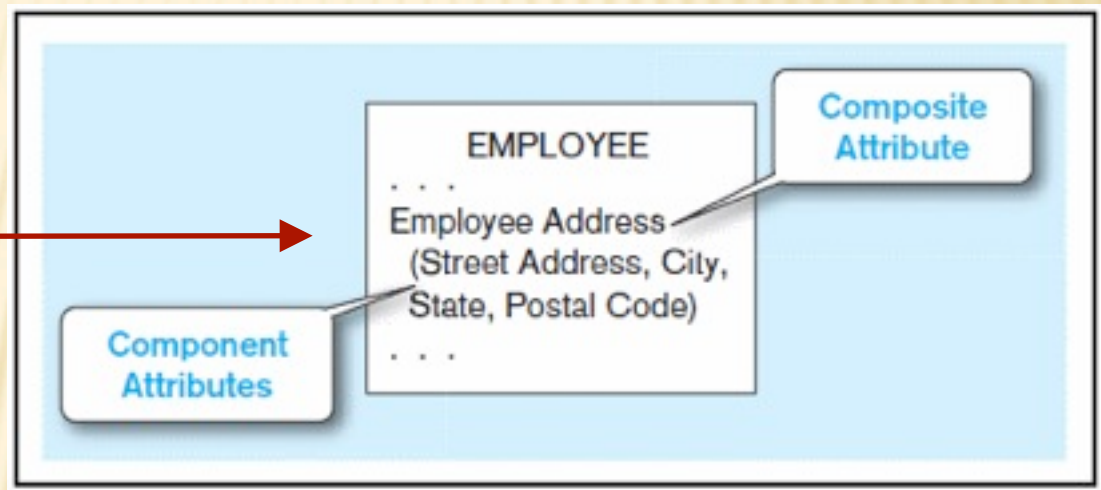


Figure 2-7 A **composite** attribute

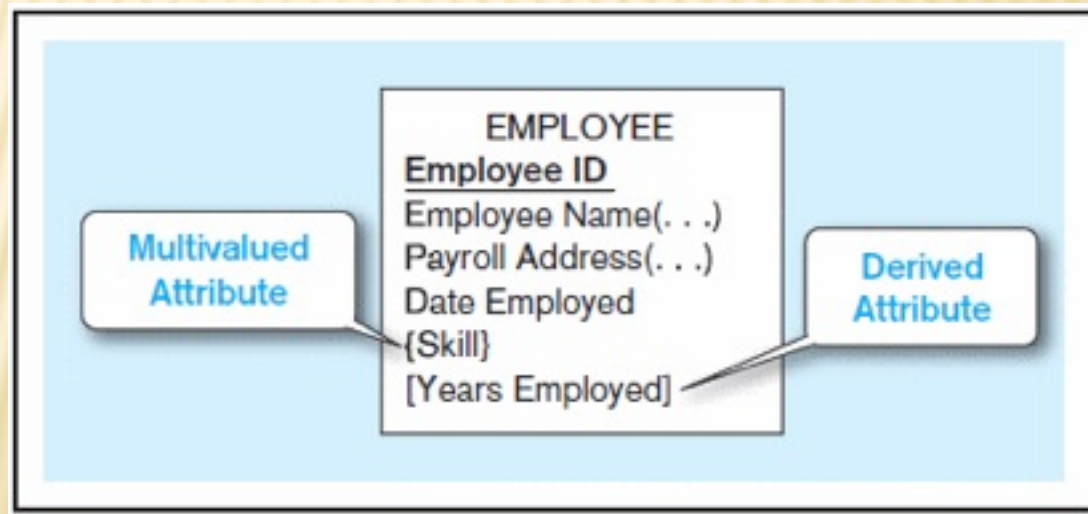
# Multi-valued and Derived Attributes

**Multivalued** – may take on more than one value for a given entity (or relationship) instance

**Derived** – values can be calculated from related attribute values (not physically stored in the database)

Figure 2-8 Entity with **multivalued** attribute (Skill) and **derived** attribute (Years Employed)

**Multivalued**  
an employee can  
have more than one  
skill



**Derived**  
Calculated  
from date  
employed  
and current  
date



# IDENTIFIERS (KEYS)

---

- ✖ Identifier (Key)–an attribute (or combination of attributes) that uniquely identifies individual instances of an entity type
- ✖ Simple versus Composite Identifier
- ✖ Candidate Identifier–an attribute that could be a key...satisfies the requirements for being an identifier

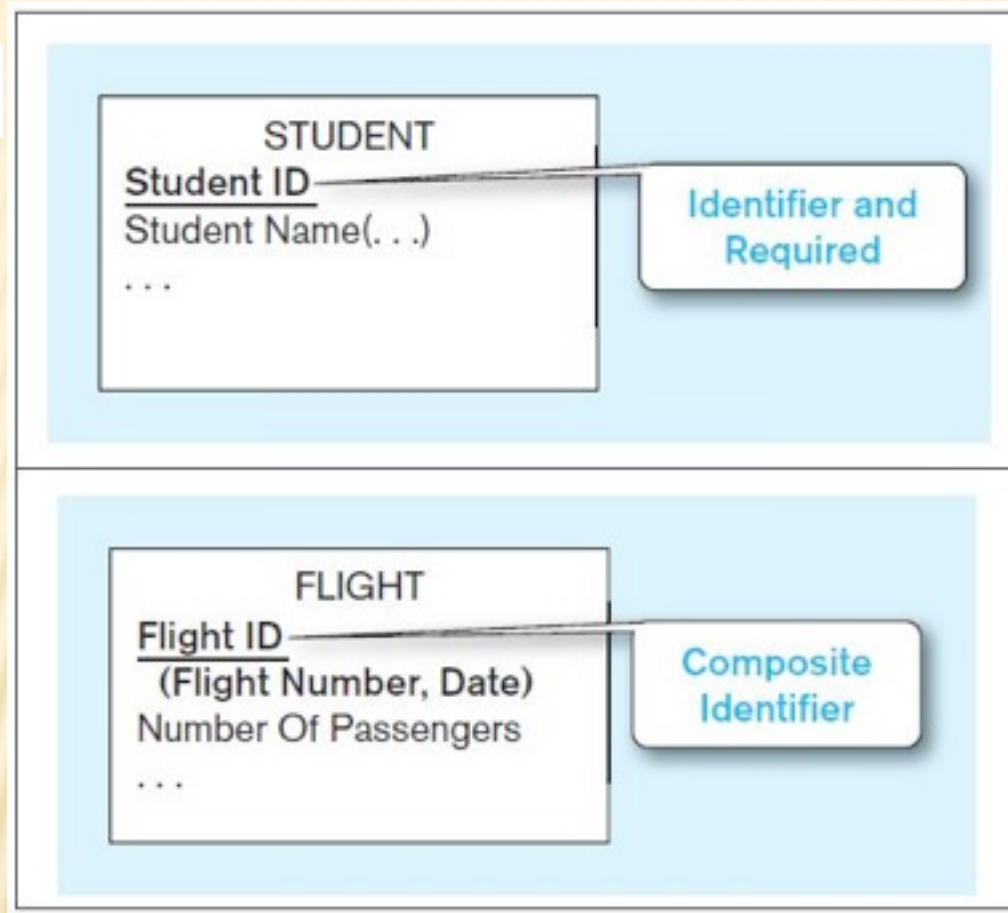
# CRITERIA FOR IDENTIFIERS

---

- ✗ Choose Identifiers that
  - + Will not change in value
  - + Will not be null
- ✗ Avoid intelligent identifiers (e.g., containing locations or people that might change)
- ✗ Substitute new, simple keys for long, composite keys

Figure 2-9 Simple and composite identifier attributes

(a) Simple identifier attribute



The identifier is boldfaced and underlined

(b) Composite identifier attribute



# NAMING ATTRIBUTES

---

- ✗ Name should be a singular noun or noun phrase
- ✗ Name should be unique
- ✗ Name should follow a standard format
  - + e.g. [Entity type name { [ Qualifier ] } ]  
Class
- ✗ Similar attributes of different entity types should use the same qualifiers

# DEFINING ATTRIBUTES

---

- ✗ State what the attribute is and possibly why it is important
- ✗ Make it clear what is and is not included in the attribute's value
- ✗ Include aliases in documentation
- ✗ State source of values
- ✗ Specify required vs. optional
- ✗ State min and max number of occurrences allowed
- ✗ Indicate relationships with other attributes

# MODELING RELATIONSHIPS

---

- ✗ Relationship Types vs. Relationship Instances
  - + The relationship type is modeled as lines between entity types...the instance is between specific entity instances
- ✗ Relationships can have attributes
  - + These describe features pertaining to the association between the entities in the relationship
- ✗ Two entities can have more than one type of relationship between them (multiple relationships)
- ✗ Associative Entity–combination of relationship and entity

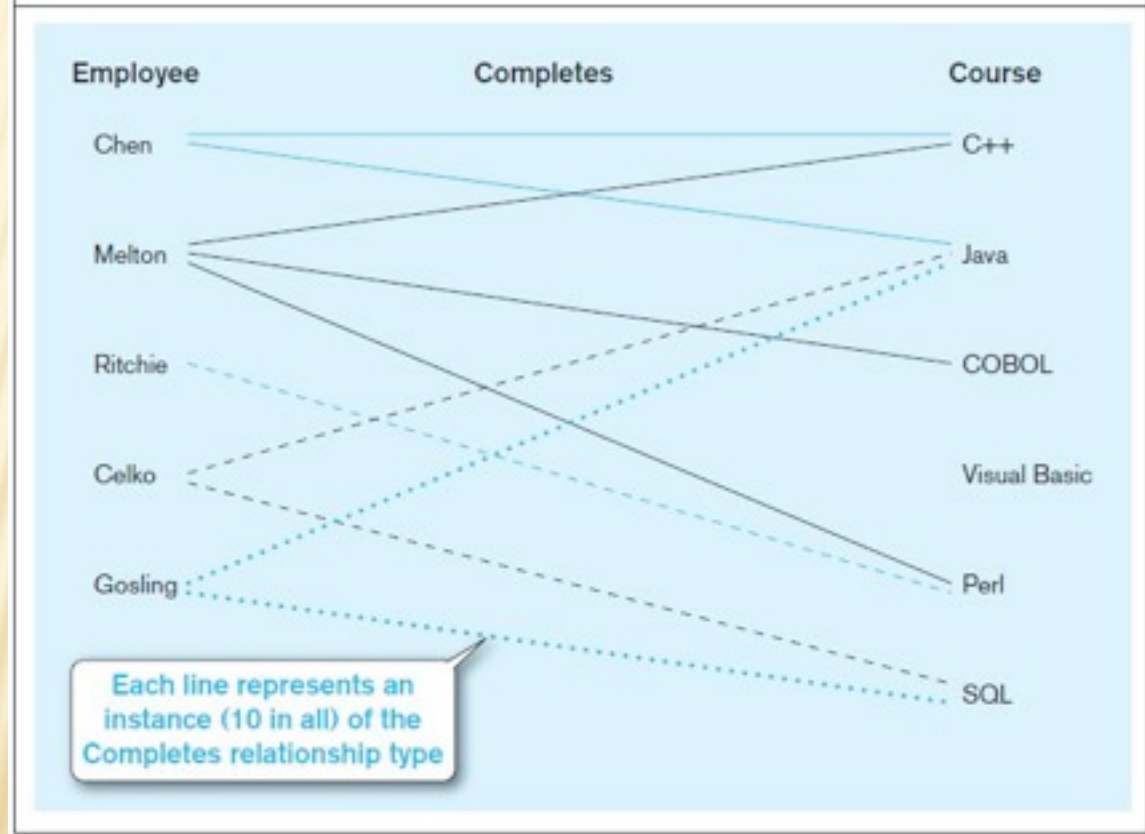


## Figure 2-10 Relationship types and instances

a) Relationship type (Completes)



b) Relationship instances

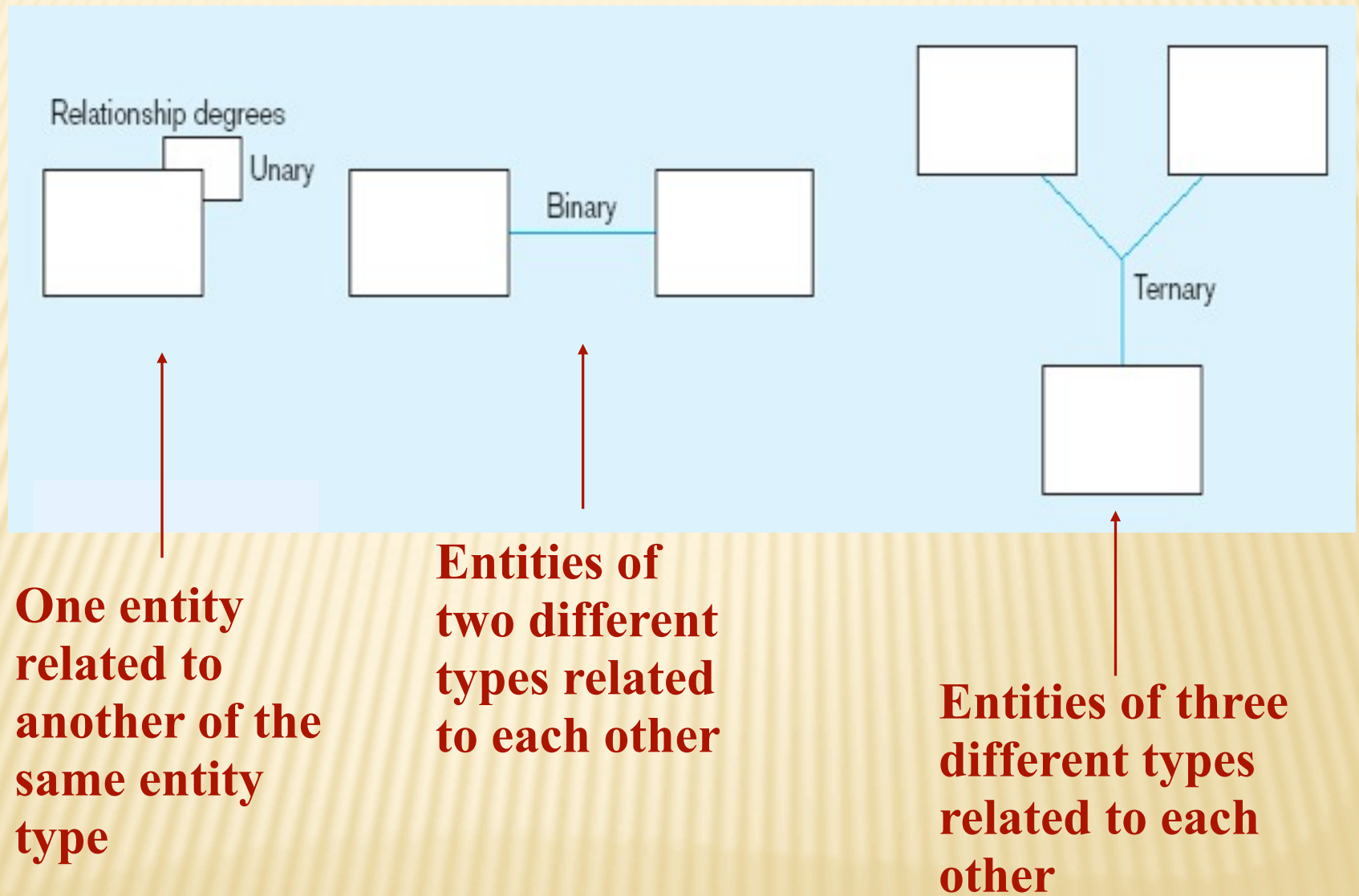


# DEGREE OF RELATIONSHIPS

---

- ✕ Degree of a relationship is the number of entity types that participate in it
  - + Unary Relationship
  - + Binary Relationship
  - + Ternary Relationship

## Degree of relationships – from Figure 2-2





# CARDINALITY OF RELATIONSHIPS

---

## ✖ One-to-One

- + Each entity in the relationship will have exactly one related entity

## ✖ One-to-Many

- + An entity on one side of the relationship can have many related entities, but an entity on the other side will have a maximum of one related entity

## ✖ Many-to-Many

- + Entities on both sides of the relationship can have many related entities on the other side

## Figure 2-12 Examples of relationships of different degrees

### a) Unary relationships

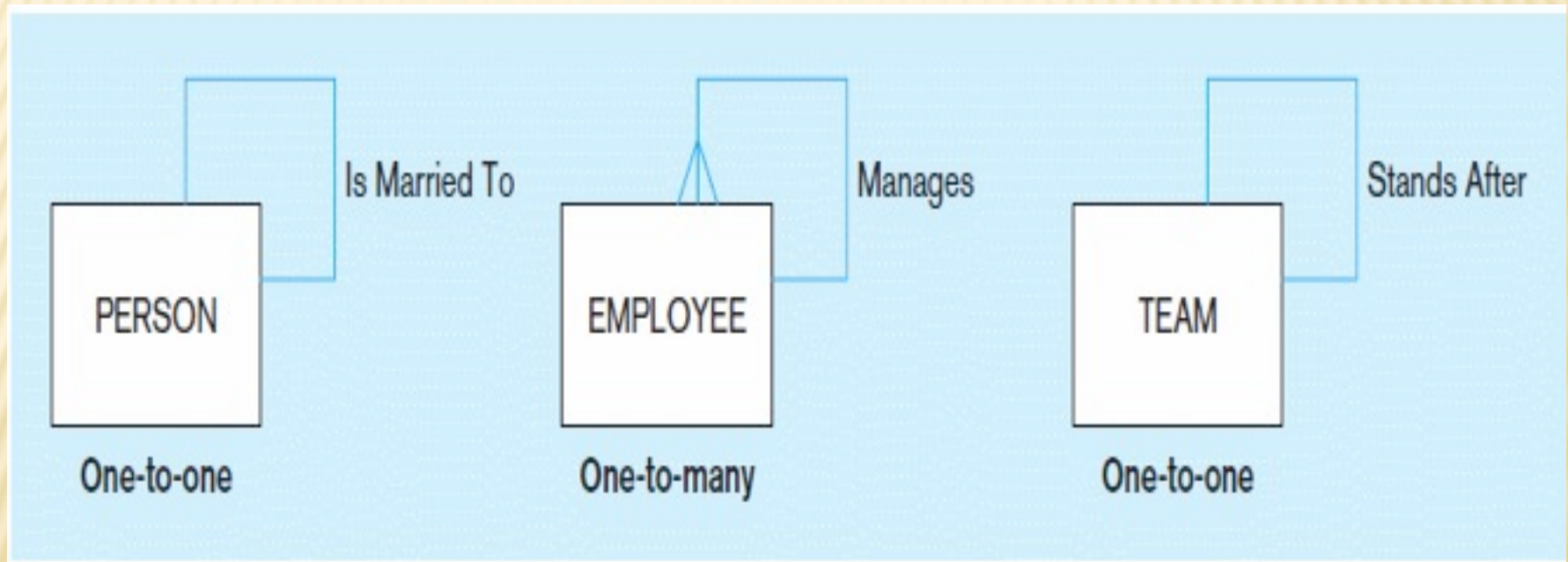
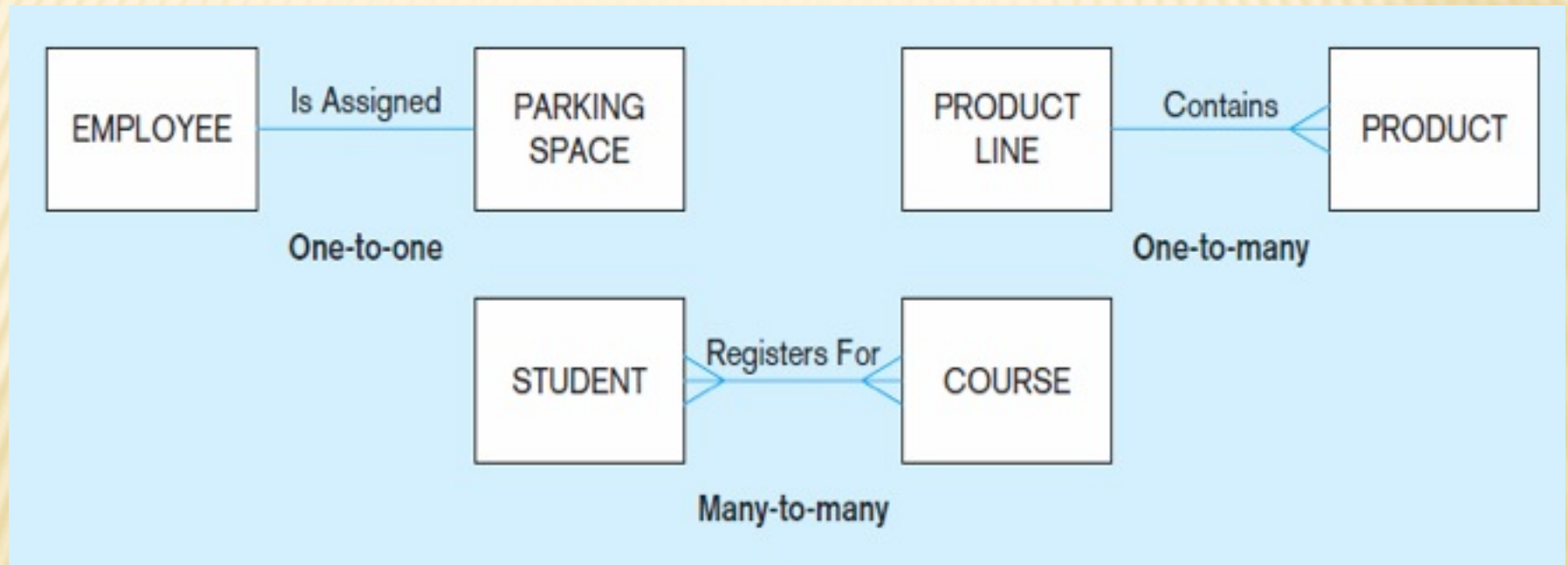


Figure 2-12 Examples of relationships of different degrees (cont.)

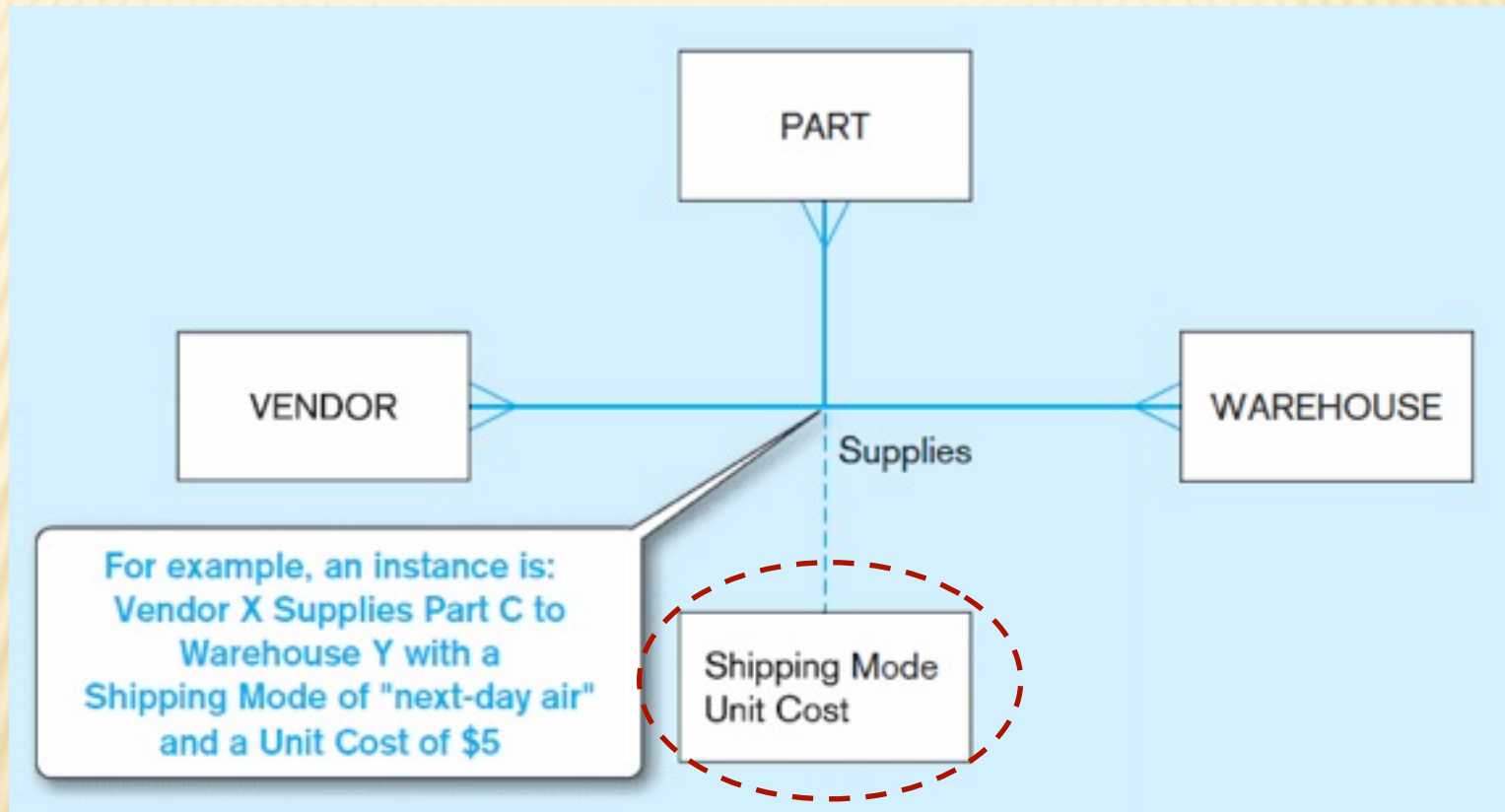
b) Binary relationships





## Figure 2-12 Examples of relationships of different degrees (cont.)

### c) Ternary relationship



**Note: a relationship can have attributes of its own**

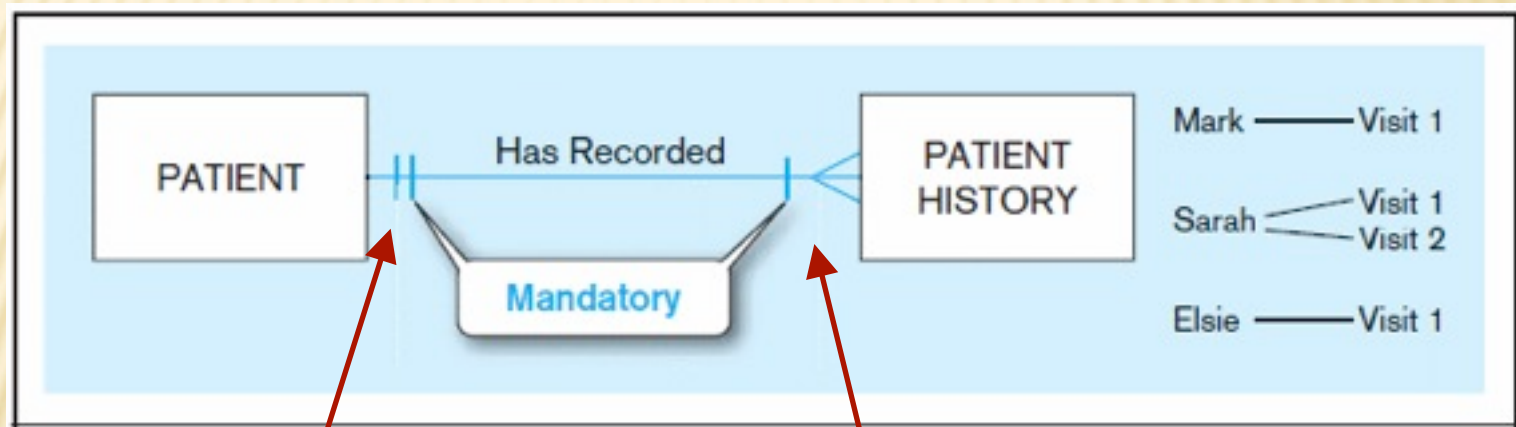
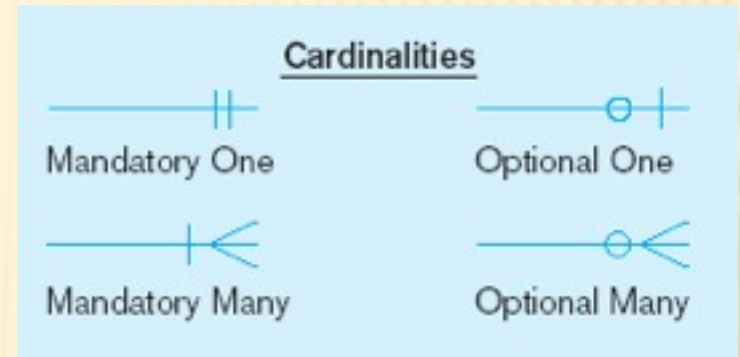
# CARDINALITY CONSTRAINTS

---

- ✗ Cardinality Constraints—the number of instances of one entity that can or must be associated with each instance of another entity
- ✗ Minimum Cardinality
  - + If zero, then optional
  - + If one or more, then mandatory
- ✗ Maximum Cardinality
  - + The maximum number

## Figure 2-17 Examples of cardinality constraints

### a) Mandatory cardinalities



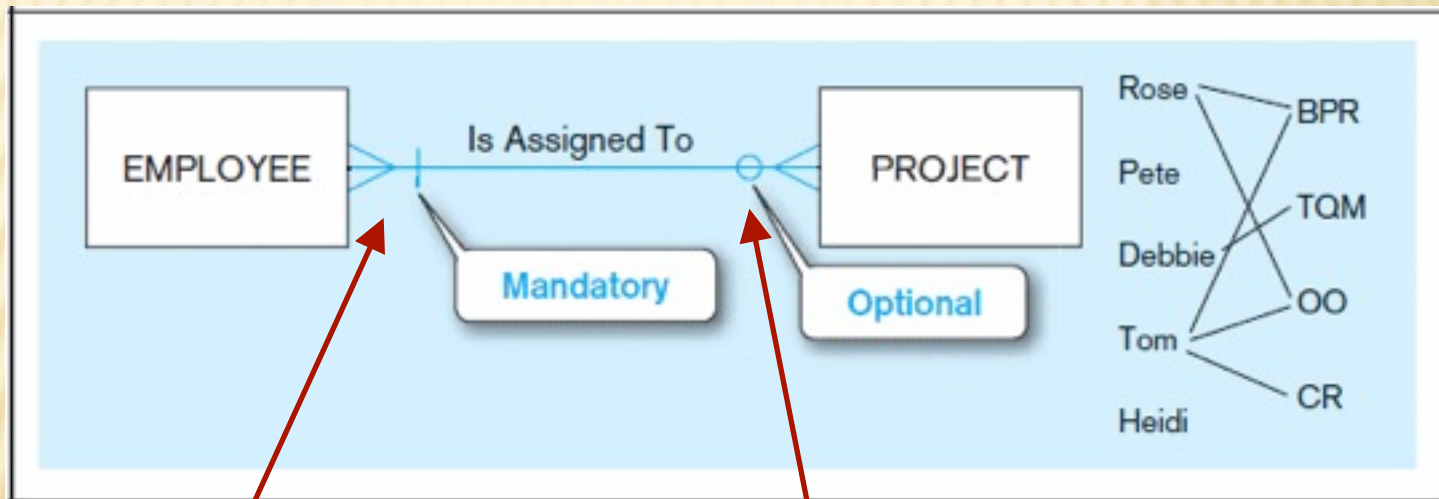
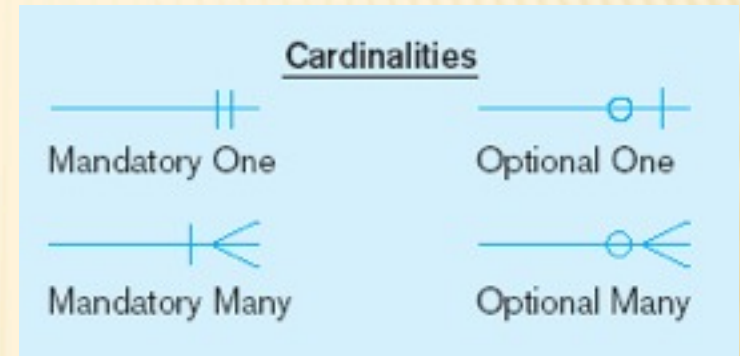
A patient history is recorded for one and only one patient

A patient must have recorded at least one history, and can have many



## Figure 2-17 Examples of cardinality constraints (cont.)

### b) One optional, one mandatory

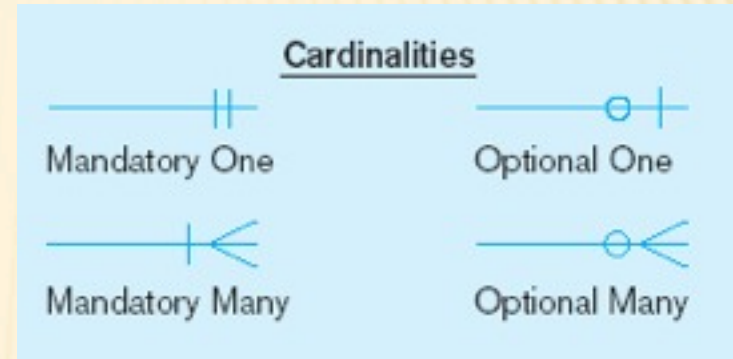


A project must be assigned to at least one employee, and may be assigned to many

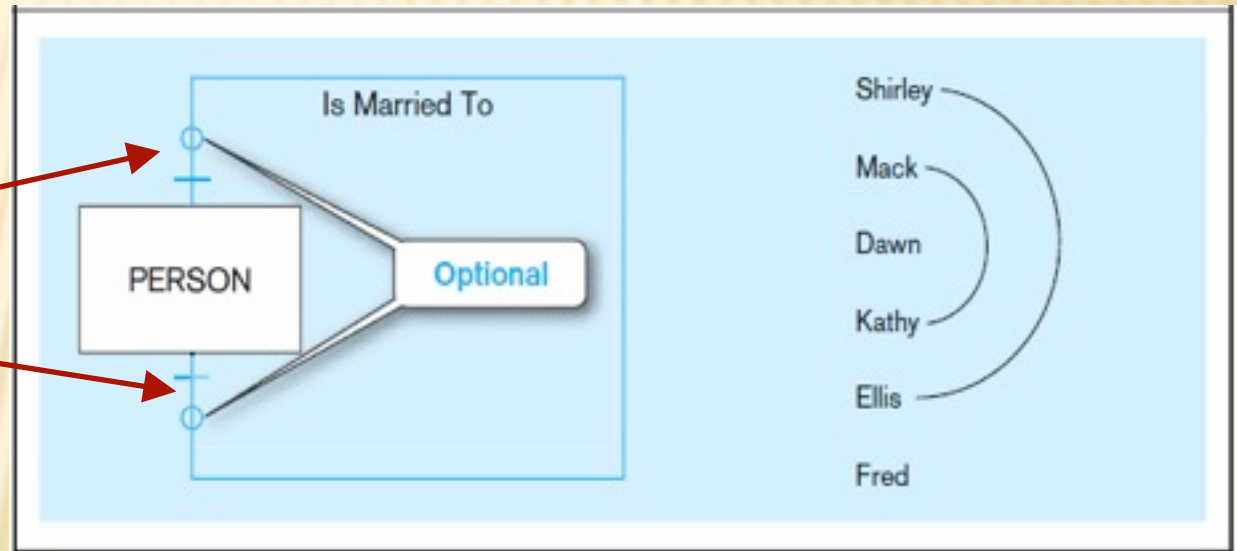
An employee can be assigned to any number of projects, or may not be assigned to any at all

## Figure 2-17 Examples of cardinality constraints (cont.)

### c) Optional cardinalities

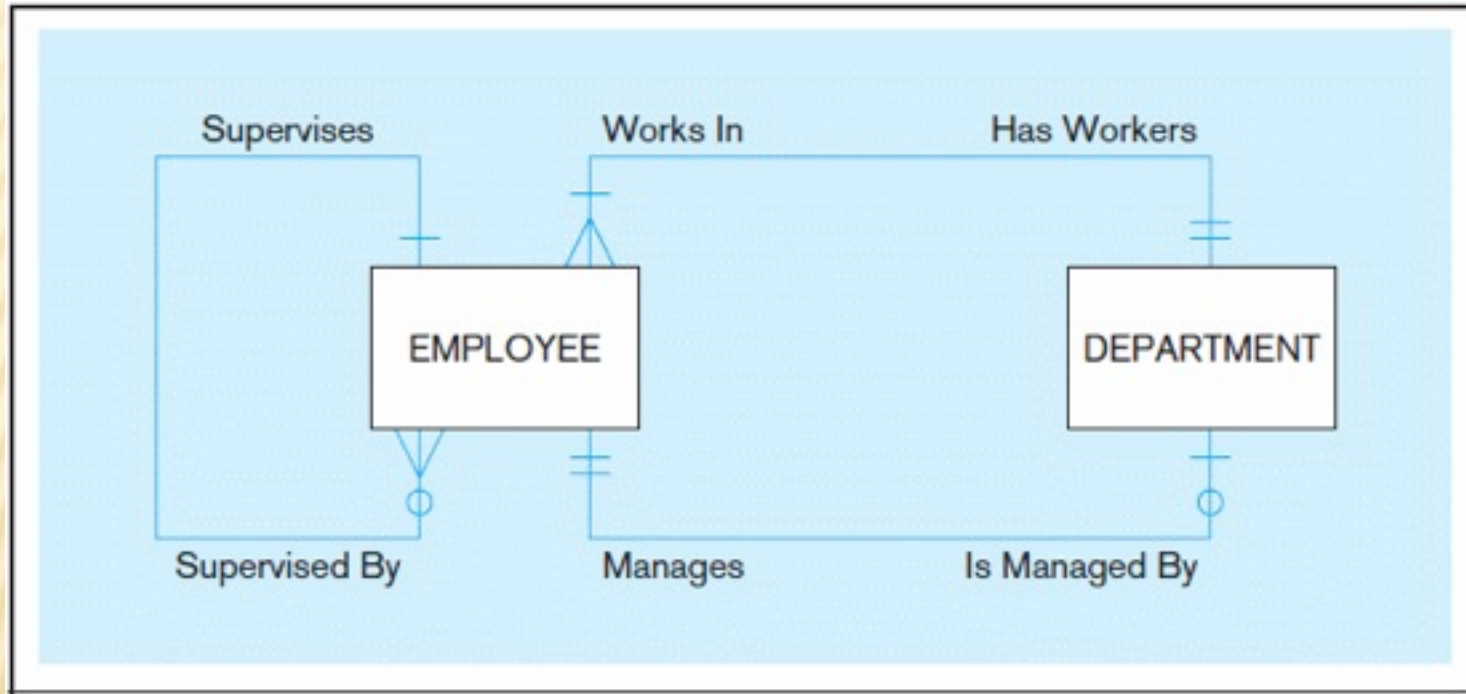


A person is married to at most one other person, or may not be married at all



## Figure 2-21 Examples of multiple relationships

### a) Employees and departments

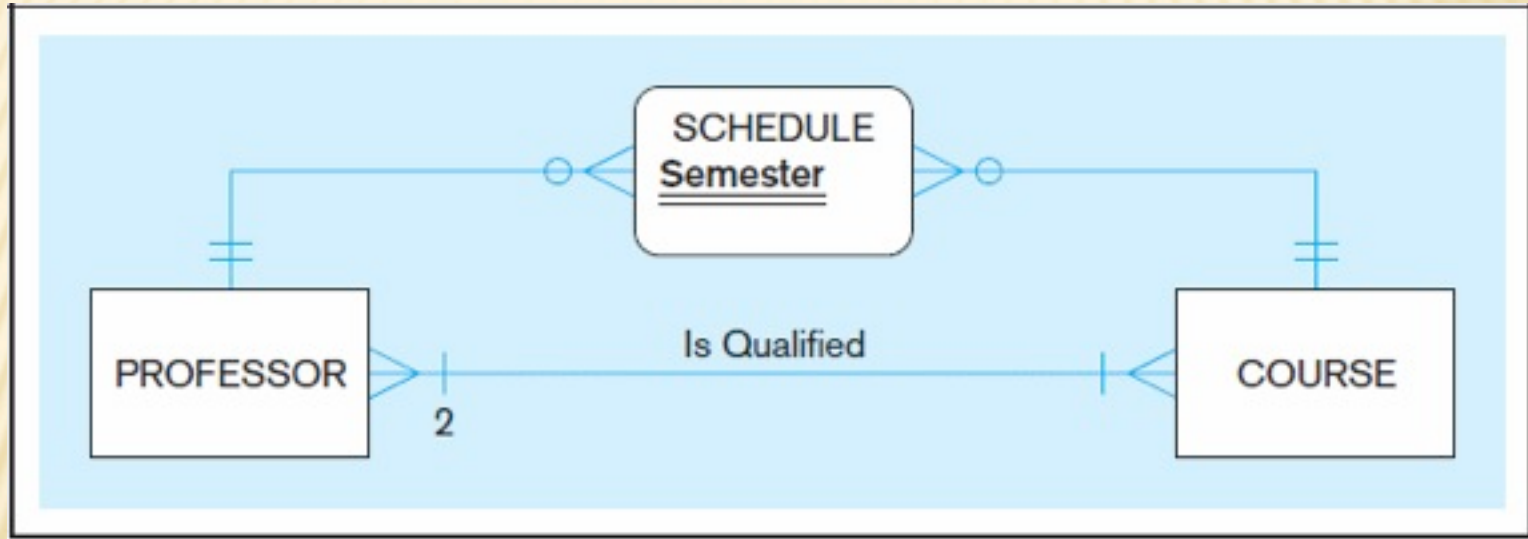


Entities can be related to one another in more than one way



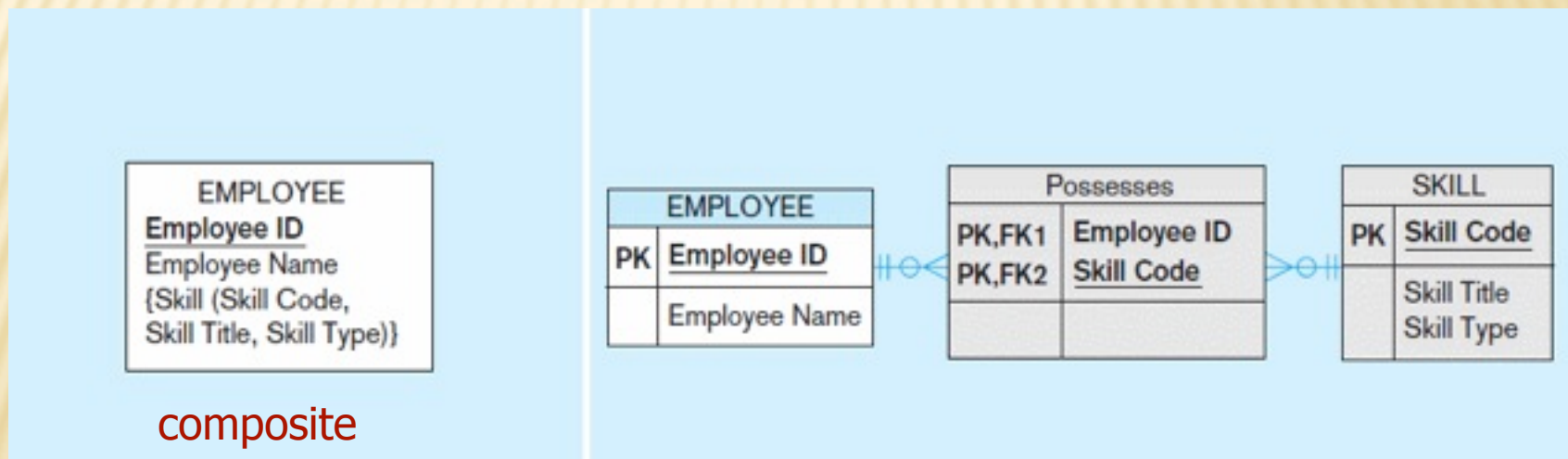
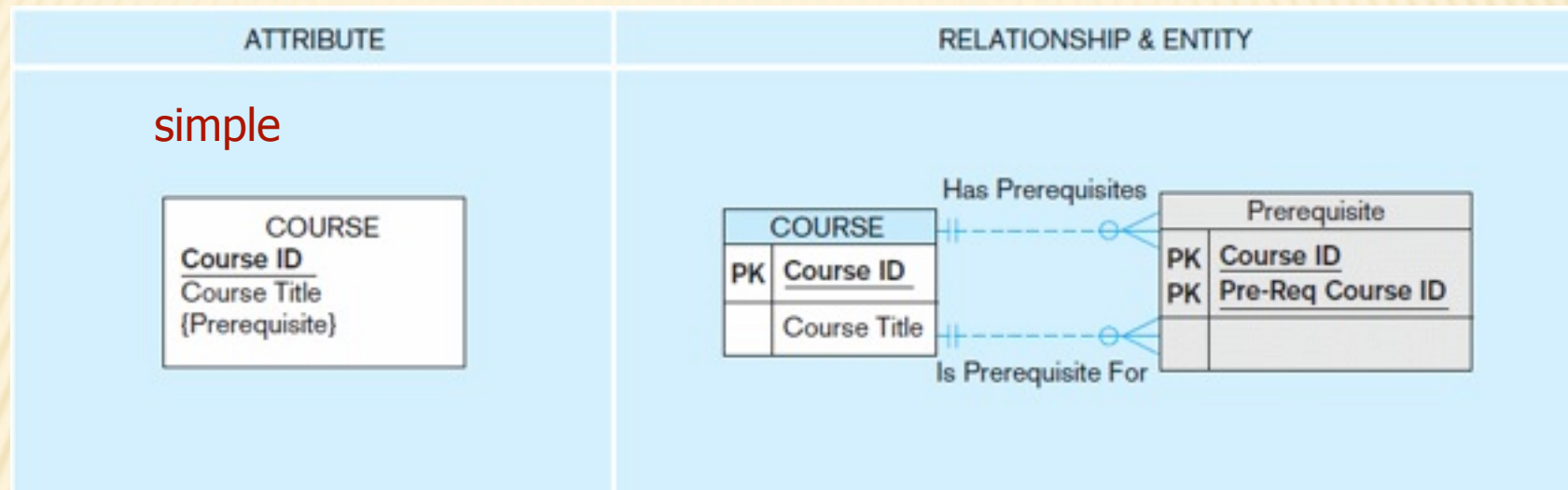
## Figure 2-21 Examples of multiple relationships (cont.)

### b) Professors and courses (fixed lower limit constraint)



Here, min cardinality constraint is 2. At least two professors must be qualified to teach each course. Each professor must be qualified to teach at least one course.

Figure 2-15a and 2-15b Multivalued attributes can be represented as relationships



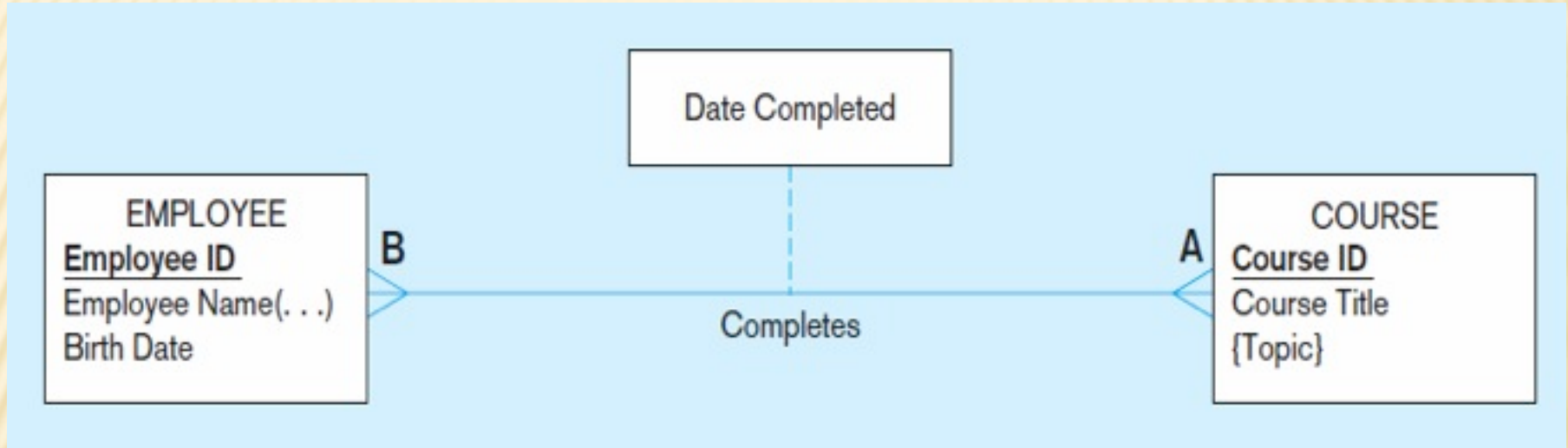
# ASSOCIATIVE ENTITIES

---

- ✖ An entity–has attributes
- ✖ A relationship–links entities together
- ✖ When should a relationship with attributes instead be an associative entity?
  - + All relationships for the associative entity should be many
  - + The associative entity could have meaning independent of the other entities
  - + The associative entity preferably has a unique identifier, and should also have other attributes
  - + The associative entity may participate in other relationships other than the entities of the associated relationship
  - + Ternary relationships should be converted to associative entities

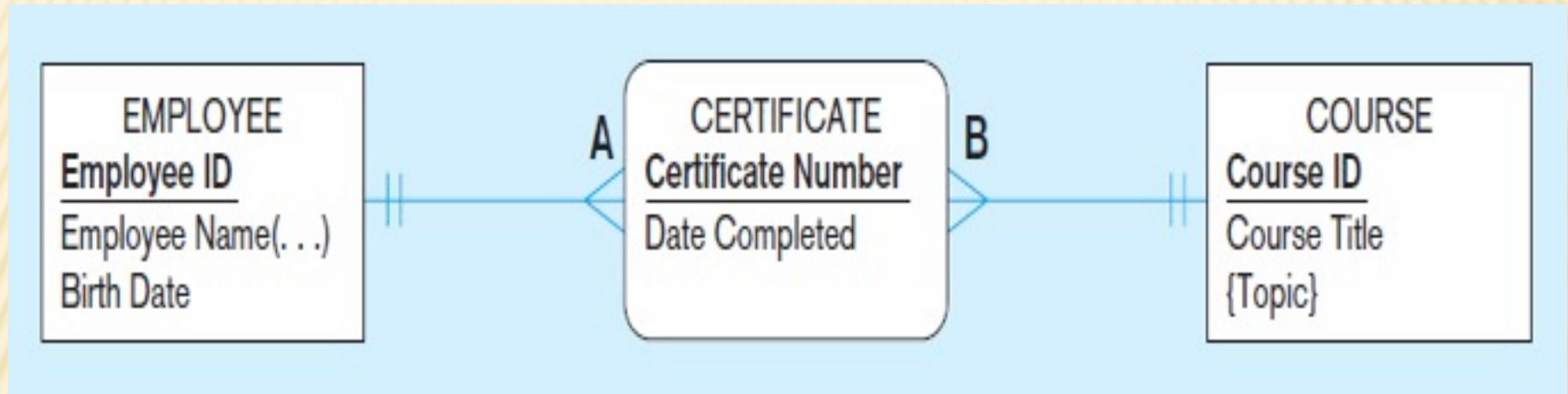


Figure 2-11a A binary relationship with an attribute



Here, the date completed attribute pertains specifically to the employee's completion of a course...it is an attribute of the *relationship*.

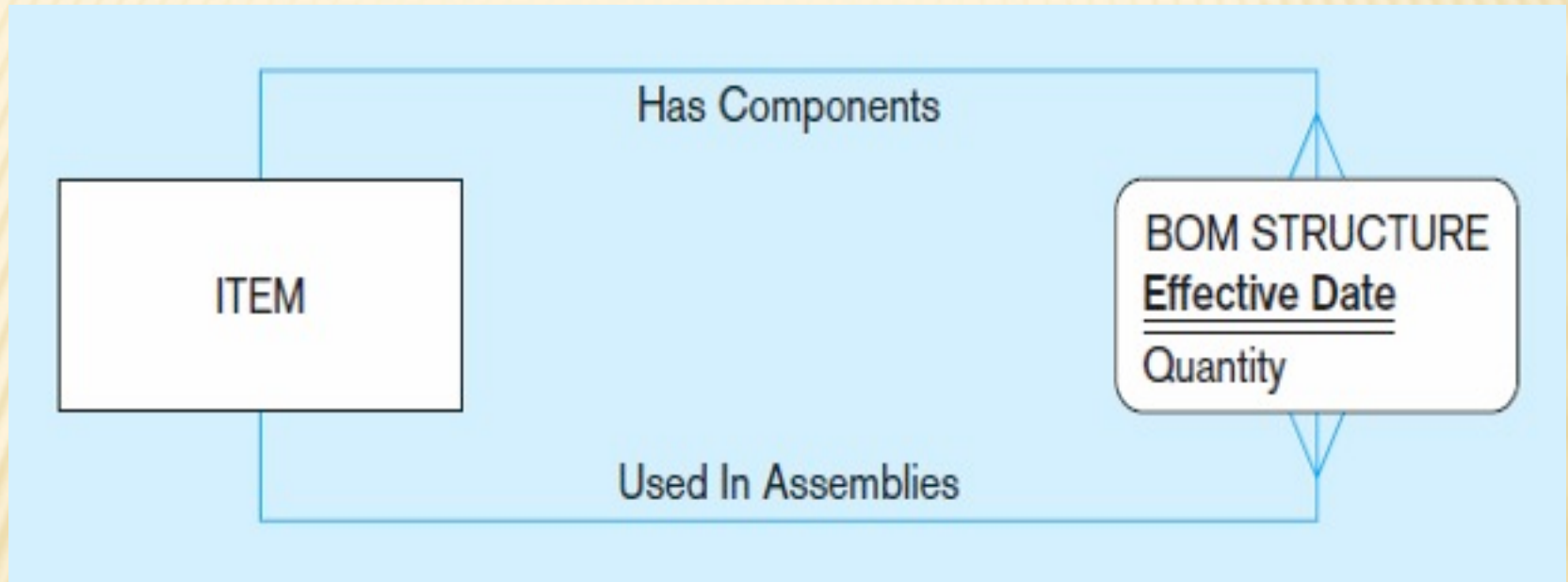
Figure 2-11b An associative entity (CERTIFICATE)



Associative entity is like a relationship with an attribute, but it is also considered to be an entity in its own right.

Note that the many-to-many cardinality between entities in Figure 2-11a has been replaced by two one-to-many relationships with the associative entity.

Figure 2-13c An associative entity – bill of materials structure



This could just be a relationship with attributes...it's a judgment call.



Figure 2-18 Cardinality constraints in a ternary relationship

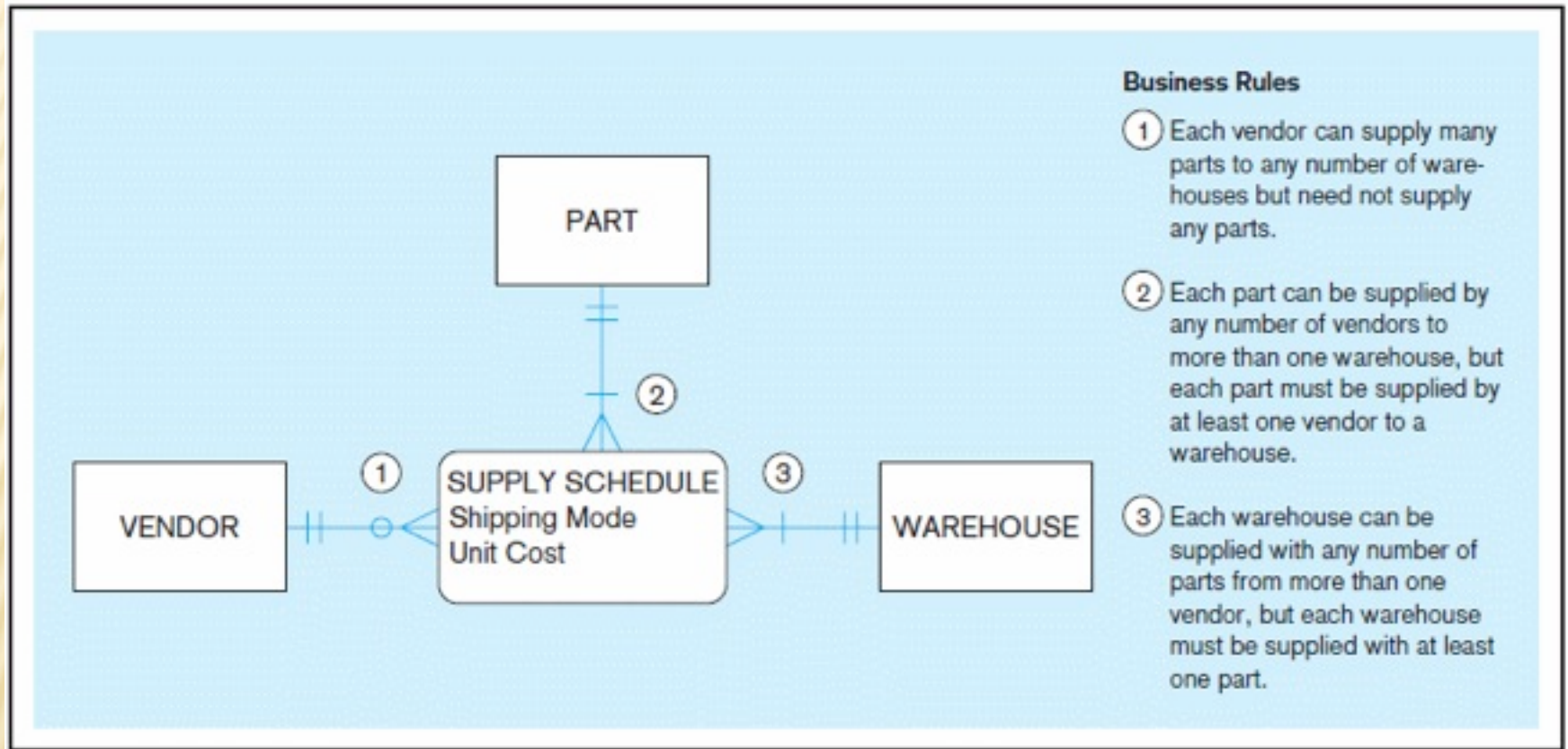
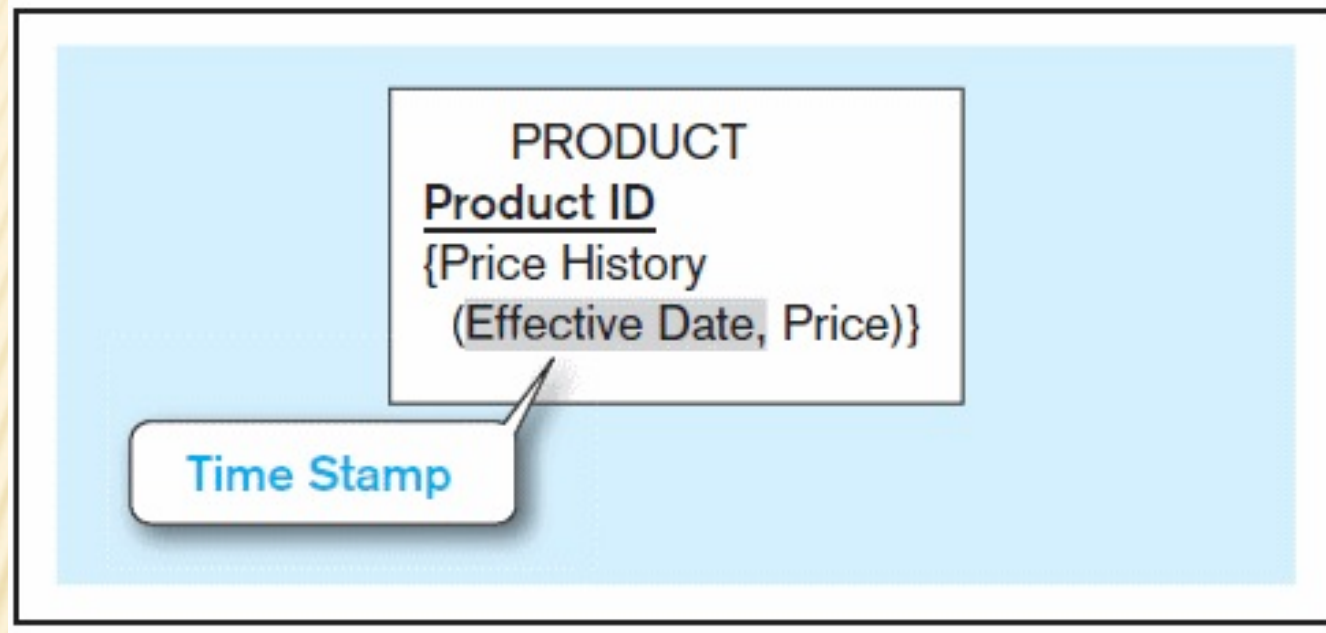


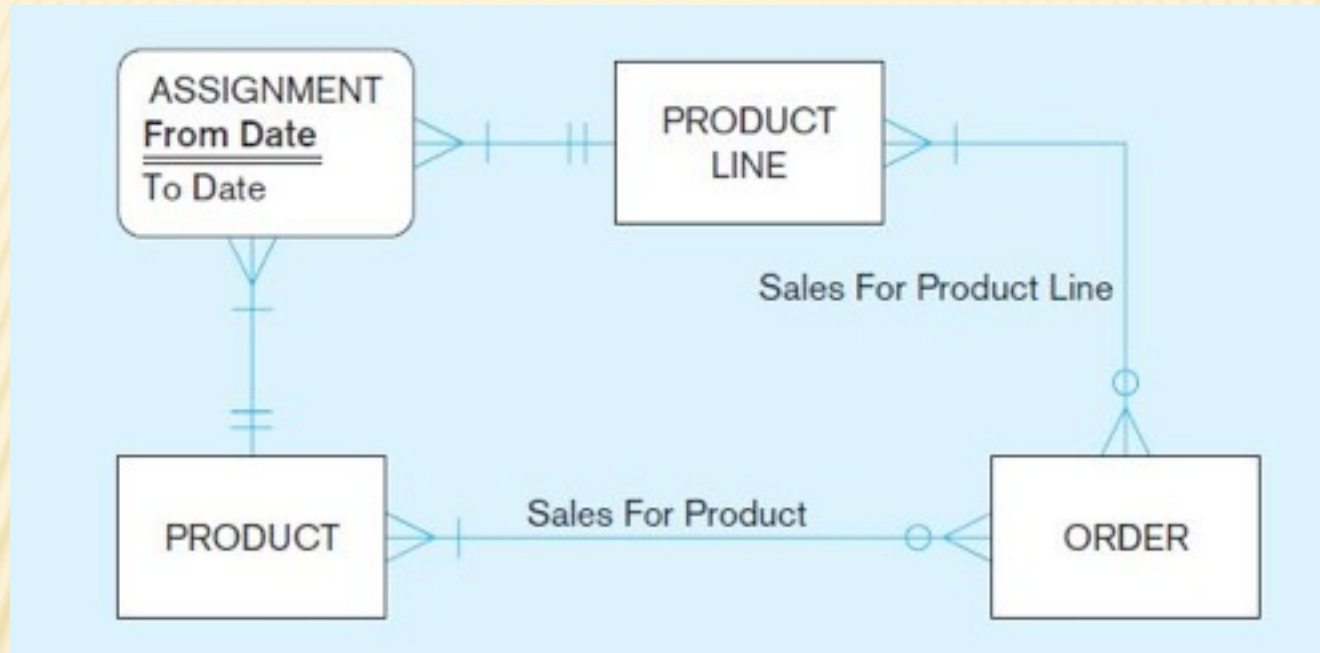
Figure 2-19 Simple example of time-stamping



**Time stamp** – a time value that is associated with a data value, often indicating when some event occurred that affected the data value

The Price History attribute is both multivalued and composite.

Figure 2-20c E-R diagram with associative entity for product assignment to product line over time



Modeling time-dependent data has become more important due to regulations such as HIPAA and Sarbanes-Oxley.

The Assignment associative entity shows the date range of a product's assignment to a particular product line.



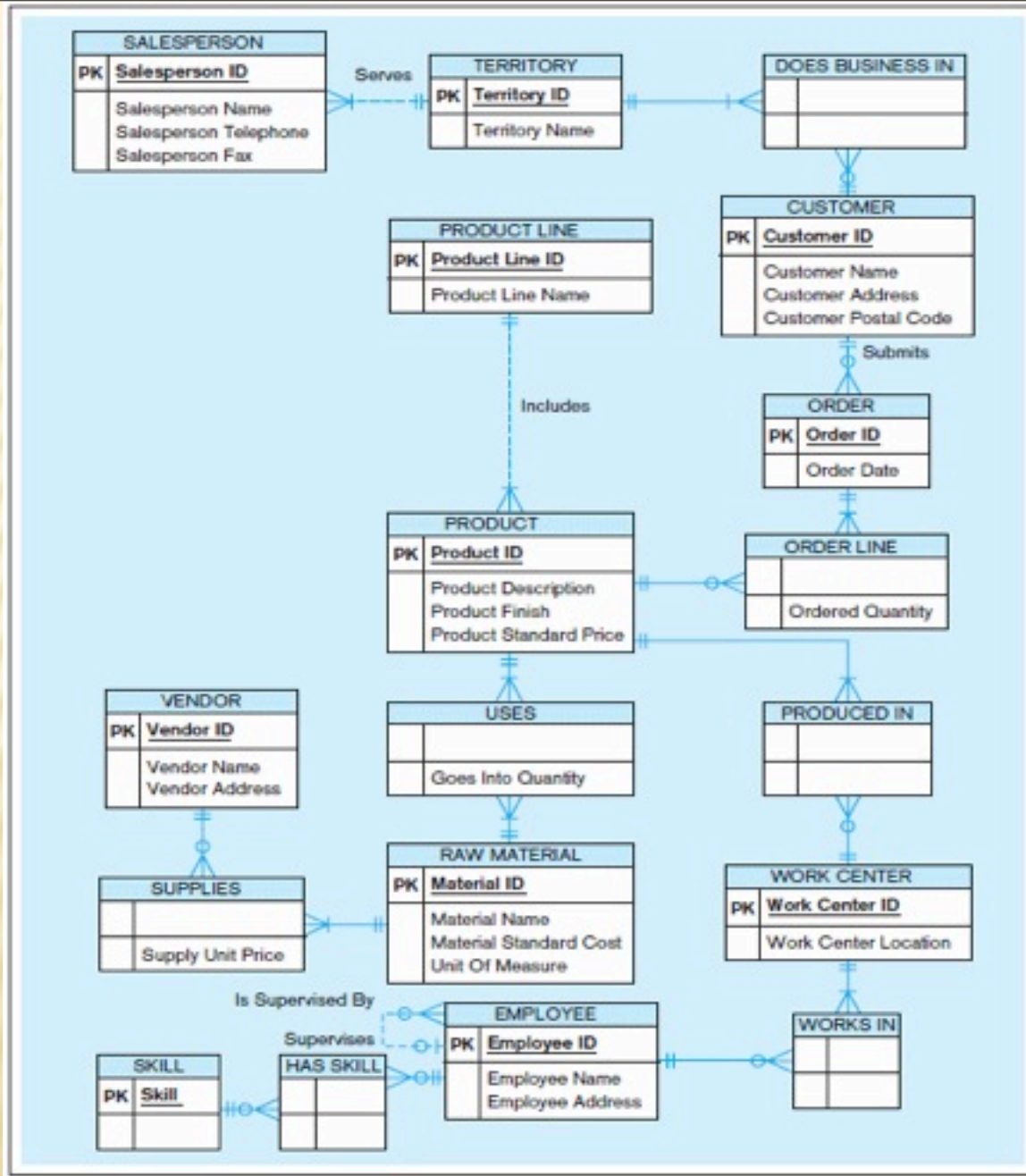


Figure 2-22  
Data model for  
Pine Valley  
Furniture  
Company in  
Microsoft Visio  
notation

Different modeling  
software tools may have  
different notation for the  
same constructs.



**This work is protected by United States copyright laws and is provided solely for the use of instructors in teaching their courses and assessing student learning. Dissemination or sale of any part of this work (including on the World Wide Web) will destroy the integrity of the work and is not permitted. The work and materials from it should never be made available to students except by instructors using the accompanying text in their classes. All recipients of this work are expected to abide by these restrictions and to honor the intended pedagogical purposes and the needs of other instructors who rely on these materials.**

Copyright © 2013 Pearson Education, Inc. Publishing as Prentice Hall